



# Software distribution in LHCb Online

Niko Neufeld, CERN



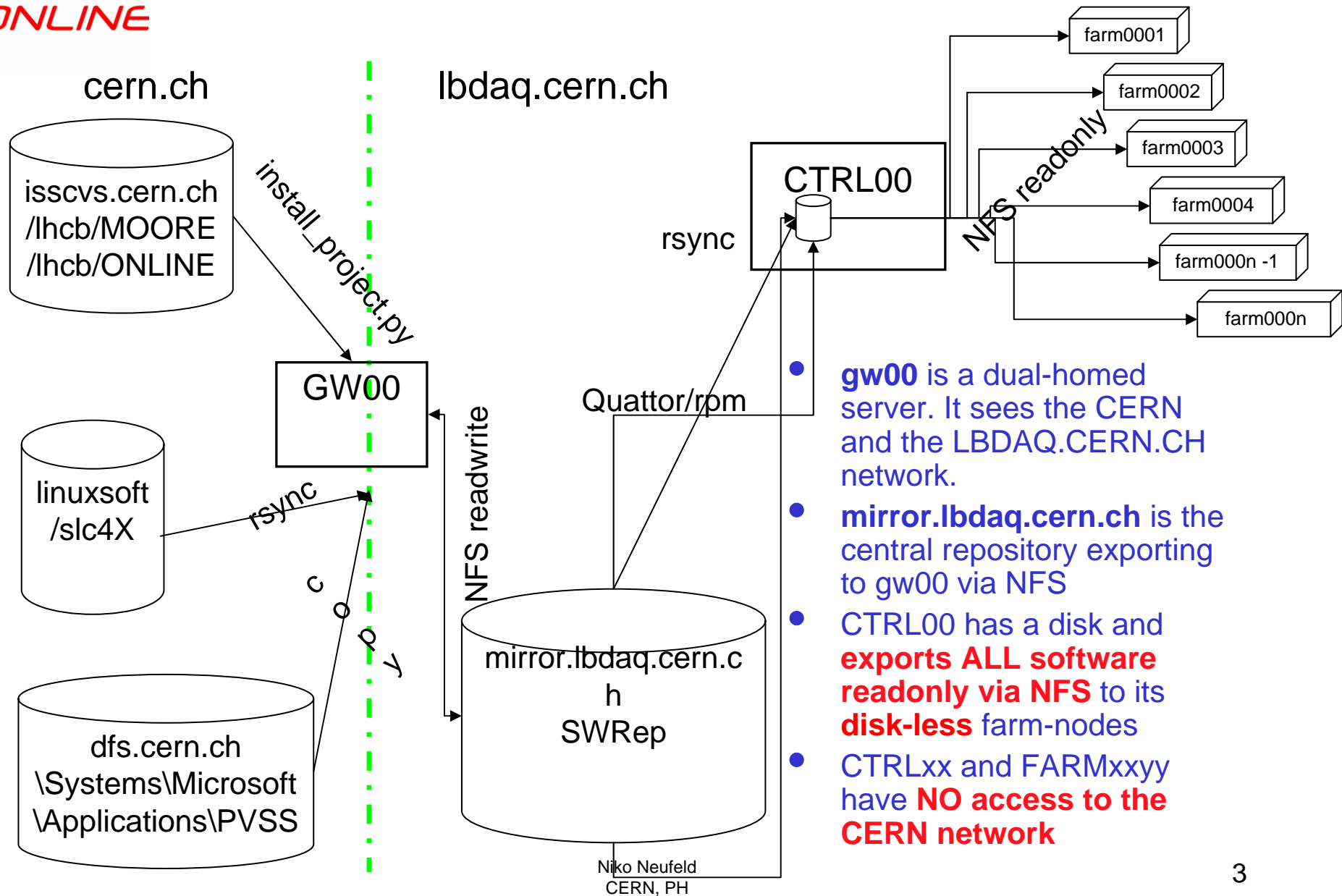
# Software in the LHCb Online<sup>(\*)</sup>

- ECS software
    - SPECS and CCPC embedded controllers
    - PVSS projects
    - Database software
    - Subdetector-specific Software
  - Farm Monitoring and Control (FMC)<sup>†</sup>
  - DAQ software
    - event-building, Data movement: Online
  - Trigger-code: Moore
  - System software: Linux
- partly using CVS/CMT  
not using Gaudi
- based on Gaudi/CMT
- based on Quattor  
(more in Loic Brarda's presentation)

\*excluding FPGA firmwares

<sup>†</sup>PC hardware monitoring and control tools with PVSS integration (functionality a la Lemon)

# Software distribution

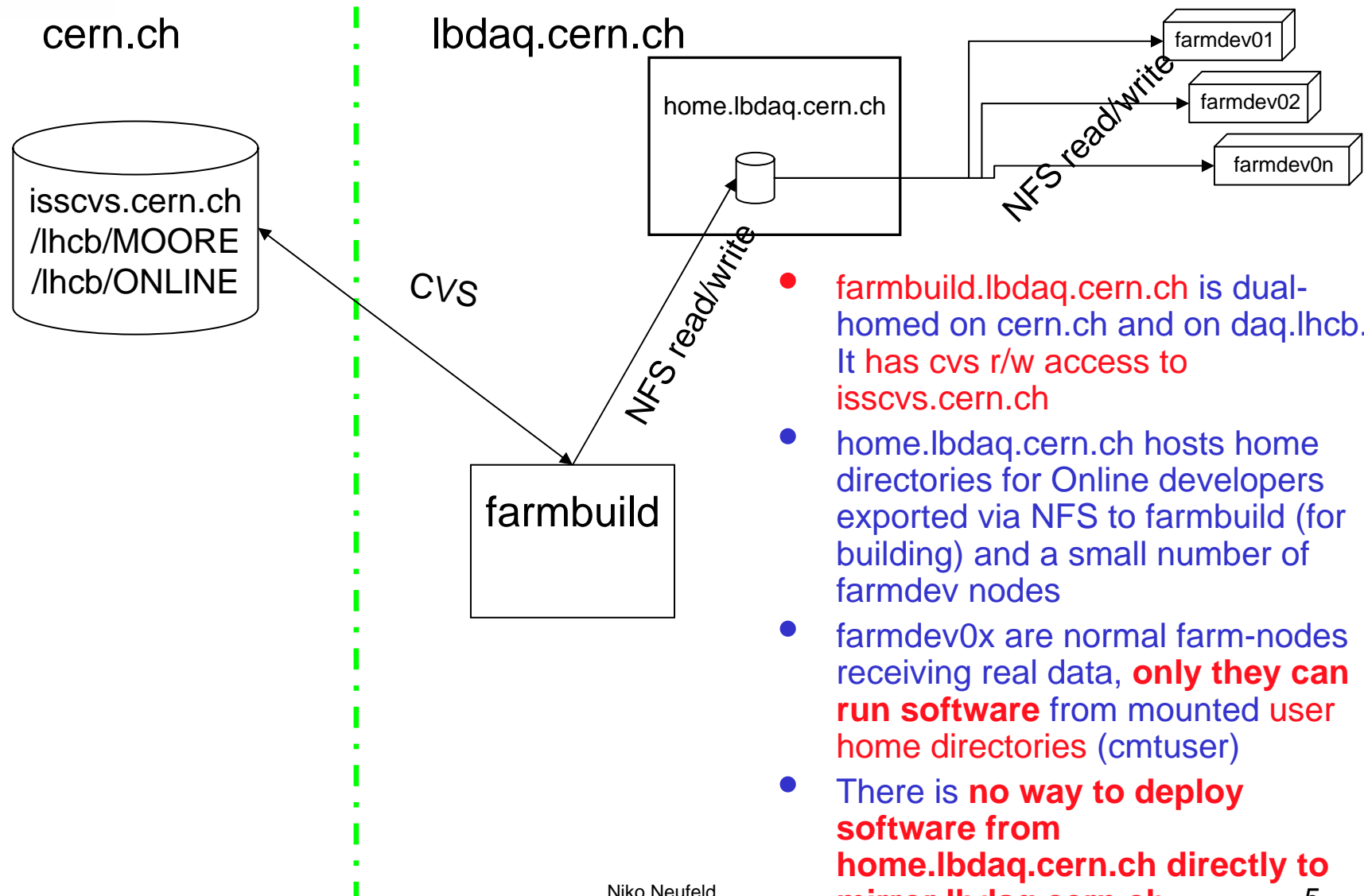


- **gw00** is a dual-homed server. It sees the CERN and the LBDAQ.CERN.CH network.
- **mirror.lbdag.cern.ch** is the central repository exporting to gw00 via NFS
- CTRL00 has a disk and **exports ALL software readonly via NFS** to its **disk-less** farm-nodes
- CTRLxx and FARMxxyy have **NO access to the CERN network**

# Tools

- `install_project.py`: standard LHCb script to install LHCb software on remote sites (GRID) from central web-repository on CERN (based on tar-balls)
- `rsync`: standard tool (will be used over ssh)
- `copy`: nothing special foreseen for Windows, a secure copy mechanism (a la `rsync/ssh`) would be nice

# Software Development



- `farmbuild.lbdaq.cern.ch` is dual-homed on `cern.ch` and on `daq.lhcb`. It has cvs r/w access to `isscv.s.cern.ch`
- `home.lbdaq.cern.ch` hosts home directories for Online developers exported via NFS to `farmbuild` (for building) and a small number of `farmdev` nodes
- `farmdev0x` are normal farm-nodes receiving real data, **only they can run software** from mounted user home directories (`cmtuser`)
- There is **no way to deploy software from `home.lbdaq.cern.ch` directly to `mirror.lbdaq.cern.ch`**

# Software deployment

- Development (if necessary) as shown in farmbuild and farmdevxx
- Back-commit to `isscvcs.cern.ch` (from farmbuild gateway)
- Tag and release in the usual form
- Automated build of tar-balls for `install_project.py`
- Update of software on Online farm via `mirror.lbdaq.cern.ch` (using `install_project.py`) and from there to the Control-PCs (== NFS-server) (using `rsync`)

# Small Open Issues

- Automatisations (tagging, tar-ball generation, syncing to mirror.lbdaq.cern.ch) needs to be improved
- “Off-line” release cycle has to be quick, otherwise the temptation to do a “quick fix” will be overpowering
- A few software packages appear in several categories, most notable: DIM
  - in CMT/Gaudi because of its use in Gaucho
  - in RPM/Quattor because of its use in the farm-control
  - in several PVSS projects
  - LD\_LIBRARY\_PATH can be used to avoid conflicts but consolidation would be nice